HSR Health Services Research

DOI: 10.1111/1475-6773.12464
METHODS ARTICLE

A Machine Learning Framework for Plan Payment Risk Adjustment

Sherri Rose

Objective. To introduce cross-validation and a nonparametric machine learning framework for plan payment risk adjustment and then assess whether they have the potential to improve risk adjustment.

Data Sources. 2011–2012 Truven MarketScan database.

Study Design. We compare the performance of multiple statistical approaches within a broad machine learning framework for estimation of risk adjustment formulas. Total annual expenditure was predicted using age, sex, geography, inpatient diagnoses, and hierarchical condition category variables. The methods included regression, penalized regression, decision trees, neural networks, and an ensemble super learner, all in concert with screening algorithms that reduce the set of variables considered. The performance of these methods was compared based on cross-validated R^2.

Principal Findings. Our results indicate that a simplified risk adjustment formula selected via this nonparametric framework maintains much of the efficiency of a traditional larger formula. The ensemble approach also outperformed classical regression and all other algorithms studied.

Conclusions. The implementation of cross-validated machine learning techniques provides novel insight into risk adjustment estimation, possibly allowing for a simplified formula, thereby reducing incentives for increased coding intensity as well as the ability of insurers to "game" the system with aggressive diagnostic upcoding.

Key Words. Risk adjustment, machine learning, regression

Risk adjustment models have become commonplace when adjusting for patient characteristics to predict clinical, cost, and quality outcomes. Typically, these models are estimated using classical linear regression (Iezzoni 2012). The trajectory of risk adjustment methodology, particularly as practiced in payment models developed by the federal government, has been largely frozen since the 1970s, failing to incorporate statistical advances that could yield more accurate formulas. This is a potentially costly oversight, as risk adjustment is extensively employed in plan payment, where it attempts to control for the impact of consumers choosing their own health plans.

For example, the new highly regulated state-level individual health insurance markets, known as Marketplaces and created by the Affordable Care Act, use risk adjustment systems for plan payment. The Marketplaces aim to provide coverage that is both affordable and comprehensive to those without health insurance. The federal government proposed a risk-adjustment formula for the Marketplaces, and all states except Massachusetts (which already had a state-level risk-adjustment formula based on its earlier health care reforms) implemented this system (Kautter et al. 2014). The Marketplace risk-adjustment models are estimated with ordinary least squares regressions and predict plan spending as a function of three sets of indicator variables: age-gender categories, diagnostic conditions, and selected diagnostic condition interactions. It is important to note that this federal risk-adjustment system was originally developed for a Medicare population. Marketplace enrollees are a fundamentally different group (e.g., much younger) from this Medicare population.

Insurers are responsible for reporting the diagnostic conditions used to calculate patient risk scores, and enrollees with more diagnostic conditions obtain larger payments from the government. This leaves insurers incentivized to code more intensely to increase revenues (Kronick and Welch 2014). Recent work by Geruso and Layton (2015) estimated that increased coding intensity in Medicare Advantage, a private coordinated care option in Medicare, has led to overpayment of $11 billion each year. Strictly fraudulent behavior, such as using prescriptions or lab results instead of physician diagnoses to code conditions, will also lead to excess payments. Both legal and illegal charge captures, often referred to as upcoding, are types of "gaming" the risk adjustment system. While regulators attempt to prevent gaming by restricting which diagnoses condition codes can be used in risk adjustment, these condition codes remain numerous and the formulas overall involve dozens of covariates in a linear regression (Kautter et al. 2014). It is therefore of interest to explore whether a simplified, more parsimonious risk adjustment formula retains the predictive performance of a larger model.

Reducing the opportunities for gaming the risk adjustment system is just one of the criteria for evaluating alternative plan payment methodologies. The primary motivation for risk adjustment of plan payments is to reduce the inefficiencies associated with adverse selection in health insurance markets (Breyer, Bundorf, and Pauly 2012). Ultimately, an alternative payment for-

---

Address correspondence to Sherri Rose, Ph.D., Department of Health Care Policy, Harvard Medical School, 180 Longwood Avenue, Boston, MA 02115; e-mail: rose@hcp.med.harvard.edu.

mula should be assessed in terms of the effects it will have on efficient sorting of consumers across plan types (Einav and Finkelstein 2011) and on ameliorating the incentives to plans to distort services to favor healthy (low-cost) enrollees over the sick (high-cost) enrollees (Glazer and McGuire 2000). While some papers have assessed these incentives in the context of employer-sponsored health insurance (Einav, Finkelstein, and Cullen 2010), Medicare (Brown et al. 2014), and Marketplaces (McGuire et al. 2014), by far the most common metric for assessing risk adjustment alternatives is simply the R^2 of the risk adjustment model (Kautter et al. 2014; van Veen et al. 2015). While acknowledging the limitations of a fit measure like R^2, as a first step in considering the potential of machine learning methods, fit measures seem like the natural place to start.

Meanwhile, the broader health statistics field is rapidly moving toward newer techniques as the era of "big data" brings increased information on patients, such as electronic health records. Given the size and complexity of these new data structures, standard statistical methods will often not be suitable or feasible. For example, it has become commonplace for there to be hundreds or thousands of covariates collected to explain an outcome of interest (van der Laan and Rose 2011). Newer statistical methods can accommodate this challenge. There is substantial potential to incorporate these advanced methods in risk adjustment. Current methods do not fully exploit the information in the data by remaining limited to parametric regression. That said, embracing more sophisticated estimation techniques with improved abilities for detecting interaction, nonlinear, and higher order effects need not indicate that a more complex risk adjustment estimator is necessarily warranted. Machine learning frameworks also allow us to screen variables, reducing a potential risk adjustment formula to, for example, just 10 variables.

The general applied statistical literature has begun to embrace these automated machine learning techniques (Lee, Lessler, and Stuart 2009; Sudat et al. 2010; Rose 2013), but this transition has yet to be made in other areas, including health economics. Machine learning methods aim to smooth over the data similarly to the way parametric regression procedures do, except they may make fewer assumptions in a nonparametric statistical model and adapt more flexibly to the data. The potential of these methods is considerable; they can provide avenues for researchers to build the exact type of interactive prediction methods they desire for use in practice. And, in the case of risk adjustment, they could lead to more accurate spending predictions. Over 50 million people in the United States are currently enrolled in an insurance program that uses risk adjustment—over three times the number in Medicare Advantage (Geruso and Layton 2015). The cost-saving implications of improved risk adjustment formulas are immense.

The contributions of this paper include introducing cross-validation ("hold-out" sampling), machine learning, and more parsimonious formulas for plan payment risk adjustment. We illustrate the use of several machine learning approaches for risk adjustment in the Truven MarketScan database (Adamson, Chang, and Hansen 2008) and assess whether use of these procedures improves risk adjustment with respect to cross-validated R^2. The core proposed framework is an ensembling machine learning technique that leverages the use of cross-validation to take a weighted average of multiple algorithms and form a single best predictor, as well as allowing for variable selection procedures that produce a parsimonious formula with a reduced set of variables. Our results demonstrated high accuracy for prediction using a small set of variables, while protecting against overfitting.

Methods

Data Source

We defined a population from the Truven MarketScan database with 2 years of continuous coverage spanning 2011–2012, which yielded 10,976,994 people. The Truven MarketScan database contains information on enrollment and claims from private health plans and employers for between 17 and 51 million people each year, and it is one of the biggest databases of this type (Adamson, Chang, and Hansen 2008). Variables available include enrollee age, sex, region, insurance plan type, date and site of service, procedures, expenditures, and inpatient diagnoses, as well as many others. We also created Hierarchical Condition Category (HCC) variables using ICD-9-CM codes, as these variables are the basis of the federal risk adjustment system (Kautter et al. 2014). This database was the source of subjects for the current Marketplace risk-adjustment models devised by the federal government. For the purposes of our work, we extracted a random sample of 250,000 people to demonstrate the proposed risk adjustment procedures. The covariates we use mimic those used in official risk-adjustment formulas, including age, sex, geographic area, five inpatient diagnosis categories, and 74 HCC variables, all from 2011. The outcome is total annual expenditures in 2012. We refer to other literature for further discussion of the database and variable construction, as well as additional approaches to sample construction (Layton, Ellis, and McGuire 2015; Rose et al. 2015).

Statistical Analysis Procedure

The vector of covariates W has length 86, and the spending outcome Y is continuous. Our goal is to develop the best predictor (i.e., the prediction function with the optimal bias-variance tradeoff). We define this formally with a loss function, which allows us to assess the performance of an estimator when applied to data. We select a loss function based on the goal of our study, which is to develop an estimator that is the best predictor of spending given a set of covariates. This can be formalized by saying we want to estimate the conditional distribution of the continuous outcome Y as a function of our covariates W. This conditional distribution is the minimizer of the squared-error loss function. Therefore, we use a squared-error loss function. As the conditional distribution we seek to estimate *minimizes* this loss function, we want our expected loss to be as small as possible to get an estimator that is close to this conditional distribution. However, the framework we discuss is flexible and can be adapted and extended for differing restrictions and loss functions to suit new metrics (Layton, Ellis, and McGuire 2015).

The estimators we consider are any machines where we can plug in our data (i.e., W) and obtain predicted values for Y. This can range from a simple local-averaging estimator to a parametric logistic regression to an advanced decision-tree algorithm. Parametric regressions are the most commonly used estimators in risk adjustment as they are easy to implement and interpret. These models make very strong assumptions that are violated in practice. The functional form of the data must be known; here that is the conditional distribution of Y given W. For the purposes of risk adjustment, our background knowledge does not support the assumptions required to a priori specify parametric regressions with confidence.

Penalized regression methods add a penalty term for the sake of reducing variance, although at a cost of added bias (Friedman, Hastie, and Tibshirani 2010). A maximum-likelihood estimator for a parametric regression is unbiased if correctly specified (which we do not expect to happen in practice), but variability can be high with collinearity. Therefore, penalized regressions offer an alternative bias-variance tradeoff. The lasso penalty, which stands for least absolute shrinkage and selection operator, offers, as the name suggests, simultaneous shrinkage of the coefficients toward zero as well as variable selection. This is because the penalty shrinks many coefficients to zero, thus eliminating those variables as contributing to the predicted values of Y. Penalized regressions can be well suited to large datasets as they prevent the overfitting that can occur due to collinear variables and high dimensionality.

However, the lasso penalty will typically select a single variable from a set of correlated variables, which may not be desirable (e.g., including only one of a set of predictors that were dichotomized from a categorical variable). Additionally, when the number of predictors is small in relation to the number of subjects, other penalties, namely the ridge penalty, will outperform the lasso penalty when variables are highly correlated. The ridge penalty offers no variable selection, as it shrinks the regression coefficients toward zero but they are never exactly zero. Other penalties are available, and general elastic nets provide some balance between these two extremes by allowing a varying degree of combined lasso and ridge penalties.

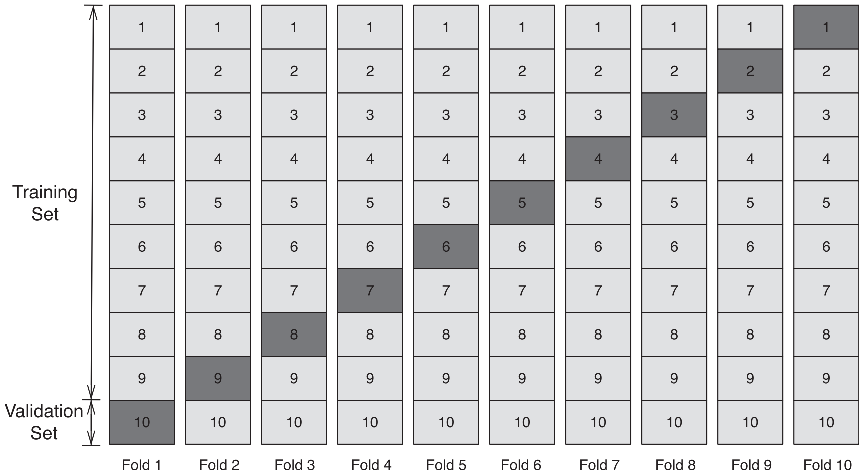
Decision tree-based methods are popular in many fields with high-dimensional data, such as genomics, and they can be useful for reducing the number of covariates and identifying more complex relationships between variables. Decision trees are developed from a set of predictor variables, similar to standard regression methods. From these predictors, the algorithm creates rules to define splits in the tree, usually with a subset of the data. The goal of the splits is to create divisions that have the most homogeneity in the outcome Y. If Y is not sufficiently homogenous after a split, the node will be split again based on another predictor variable. Otherwise, it will be defined as a terminal node and assigned an outcome value. Different decision tree methods employ varying techniques to grow the tree, and common procedures grow very large trees (i.e., many nodes) with a backwards deletion step used at the end to discover the optimal tree size and remove terminal nodes (Breiman et al. 1984). Overfitting is typically an issue with decision trees, especially when the tree has a large number of terminal nodes. This overfitting can lead to poor performance when the decision tree is applied to the full data or another dataset from a similar population. Random forests is an ensembling method that grows many trees in an effort to protect against outlier trees and overfitting, although overfitting can still be an issue. The algorithm uses a subset of the data to define the splits in the tree, but unlike in single decision trees, random forests takes a bootstrap sample for each tree. The unselected observations are used to validate the procedure. Once a large number of trees have been produced (often 500 or more), final rules are developed based on the modal or average value across the trees.

Artificial neural networks, also referred to as neural nets and recently rebranded as deep learning, are algorithms that attempt to explain an outcome given a set of input variables by postulating a series of interconnected nodes within multiple layers (i.e., the network). The relationships between the interconnected nodes are defined by weights, calculated with respect to one of a number of different rules. The algorithm starts with an initial guess for the weights of the nodes, and then iterates, adjusting the weights given how well it did in predicting the outcome. The algorithm was inspired by the complex relationships of neurons within the brain. We guide the interested reader to additional literature for further details of this complex technique (Venables and Ripley 2002).

There are many other possible algorithm choices available (Friedman, Hastie, and Tibshirani 2001; James et al. 2013), but even considering only regression, lasso penalized regression, ridge penalized regression, a balanced elastic net, a single tree, random forests, and a neural net, it is not clear which procedure will yield the best performance. If we want to identify the single best algorithm from among these choices, we could employ cross-validation. Cross-validation involves the use of rotating "hold-out" samples from within our data to assess the performance of an algorithm. It allows us to assign measures of performance to each algorithm that reflect how the procedure would behave in practice. If the algorithm is only effective at producing accurate estimates in the data used to fit the algorithm, it is not useful to employ in another setting with novel data, which is the goal of most risk-adjustment problems.

The utility of cross-validation is easy to understand once the procedure is illuminated. We discuss 10-fold cross-validation here, as it has many desirable statistical properties and low computational burden compared to other types of cross-validation (Dudoit and van der Laan 2005; van der Laan, Polley, and Hubbard 2007). Our sample data are partitioned into 10 mutually exclusive blocks of equal size. In the first "fold," we isolate the first nine blocks to serve as the training set, so-called because we will fit each of our five algorithms discussed above on this set of data, with the last block serving as a validation set. After each of our algorithms is "trained" using the data in the training set in fold 1, this fit is used to generate predicted values for the observations in the validation set. Thus, after fold 1 is completed, we only have predicted values for 1/10th of the data for each algorithm, but these values were generated on data not used to fit the algorithm, thus protecting against overfitting. The validation set rotates such that each block serves as the validation set once. See Figure 1. At the end of the complete 10-fold cross-validation procedure, we have predicted values for each observation using the held-out validation sample from each fold. A cross-validated mean squared error for each algorithm can then be calculated using these predicted values:

Figure 1: Visualization of Ten-Fold Cross-Validation



$$CV\ MSE = 1/n \sum\nolimits_{i=1}^{n} (Y_i - \hat{Y}_i)^2$$

where $\hat{Y}$ represents the predicted probabilities for a particular algorithm and n the sample size. The algorithm with the smallest cross-validated mean squared error is the optimal choice given our loss function discussed earlier. A cross-validated R^2 can also be calculated:

$$CV\ R^2 = 1 - \left(\sum\nolimits_{i=1}^{n} (Y_i - \hat{Y}_i)^2 / \sum\nolimits_{i=1}^{n} (Y_i - \bar{Y}_i)^2 \right)$$

where $\bar{Y}$ is the mean of Y. The optimal choice algorithm has been referred to as the cross-validation selector or the discrete super learner (Dudoit and van der Laan 2005; van der Laan, Polley, and Hubbard 2007).

One might query whether it is possible to improve on this method for selecting the optimal algorithm. Recall that a single decision tree is often improved by using the ensembling random forests procedure. It would be natural to then consider a general ensembling framework that allows us to average across many different types of algorithms. Thus, if multiple algorithms capture important but unique components of the prediction function, our final prediction function will incorporate all of them. This is exactly the method we propose: an ensembler that takes a weighted average of multiple algorithms to produce a single combined algorithm with optimal mean squared error. This machine learning approach is called super learning, and it has been developed and applied in the statistics literature (van der Laan, Polley, and Hubbard

2007; van der Laan and Rose 2011). One can also conceptually view the ensembling super learner as taking a weighted average of the predicted values from each algorithm considered. The estimator requires only a few additional steps beyond those described in the 10-fold cross-validation procedure.

Given that we have already performed 10-fold cross-validation and obtained predicted values in the validation sets for each observation and algorithm, we will now use these values to calculate the optimal weight coefficients. This takes the form of a regression of *Y* on these predicted values, with a separate column of predicted values for each algorithm. The optimal weight coefficients are the coefficients in front of each of these column variables in the regression. One can show that these optimal weights minimize the cross-validated risk (van der Laan, Polley, and Hubbard 2007; van der Laan and Rose 2011). The penultimate step is to fit each algorithm with the full data and combine these fits with the weights to generate the super learner prediction function. The super learner prediction function is thus now defined as a weighted combination of algorithms. Some algorithms will typically receive a weight of zero and are therefore ignored for the purposes of generating predicted values. To produce final predicted values for the full data, one feeds the data through the described super learner function. For example, if the lasso penalized regression had a weight of 0.50 and the random forests algorithm also had a weight of 0.50, with all other algorithms receiving a weight of zero, the super learner predicted values would be a weighted combination of the lasso penalized regression predicted values and the random forests predicted values. Specifically, you would multiply the predicted values generated by the lasso by 0.50 and add them to the predicted values generated by the random forests procedure, also having been multiplied by 0.50, to produce your final predicted values. To apply the super learner function to a new dataset, one would use the fixed fits of the lasso and random forests procedure established by the original dataset, as well as the weights, running the new data through this function. Additional specifics regarding the mechanics of the super learner can be found in other literature (van der Laan, Polley, and Hubbard 2007; van der Laan and Rose 2011).

There is another key property of the super learning framework that will be particularly important in the risk adjustment setting. With high-dimensional data, it can be useful to reduce the number of variables considered for adjustment, thus simplifying the final formula. In super learning, a screening step can be included within the overall algorithm and its cross-validation. We use a random forests screening step that takes the top 10 variables with the highest variable importance measures. These

Table 1: Several Machine Learning Methods for Risk Adjustment

Algorithm	*Description*
Parametric regression	Main terms parametric linear regression
Lasso	Penalized regression; shrinks some covariate coefficients to zero, eliminating their contributions to the predicted outcome
Ridge	Penalized regression; shrinks some covariate coefficients toward zero, but does not eliminate any covariates by shrinking to zero
Elastic net	Penalized regression; allows various penalties combining ridge and lasso penalties
Neural net	Iterative weighted nodes in a network
Single tree	Recursive partitioning and regression tree
Random forests	Decision tree-based method; uses bootstrapping to aggregate
Super learner	Ensembling method; takes a weighted average of included algorithms to produce a single best prediction function
Discrete super learner	Ensembling method; selects the single algorithm with the best performance

variable importance measures are calculated by leaving each variable out of the dataset and comparing the decision tree results to when it is retained within the data. Alternative screening step methods include using univariate regression to select the top 10 smallest p-values or using lasso and keeping all variables with nonzero coefficients. Once the screening step selects the reduced set of variables, only these variables are used within the individual algorithms.

We consider the seven individual algorithms discussed in this section, seven additional versions of them with the random forests screening step (for a total of 14 individual algorithms), as well as the discrete super learner and the super learner. See Table 1 for a summary of methods. Analyses were performed in the R programming environment with packages glm, glmnet, rpart, randomForest, nnet, and SuperLearner (Liaw and Wiener 2002; Venables and Ripley 2002; Friedman, Hastie, and Tibshirani 2010; Polley and van der Laan 2013; R Core Team 2013; Therneau, Atkinson, and Ripley 2013).

RESULTS

Truven MarketScan Database

The variables in our dataset are summarized in Table 2. The 10 variables retained by the random forests screening step were age category 21–34 years,

Table 2: Characteristics of Truven MarketScan Sample (n = 250,000)

Variable		
Total annual expenditures in second year, mean (SD)	\$5,476	(\$17,736)
Male	119,736	48%
Age, years		
$20 < x \leq 34$	59,318	24%
$34 < x \leq 54$	134,664	54%
Region		
Northeast	48,483	19%
Midwest	71,126	28%
South	93,331	37%
Metropolitan statistical area	211,320	85%
Inpatient diagnoses		
Heart disease	5,586	2%
Cancer	2,557	1%
Diabetes	2,819	1%
Mental health	5,395	2%
Other	22,108	9%
Hierarchical condition categories		
Metastatic cancer	391	0.16%
Stem cell transplant/complication	31	0.01%
Multiple sclerosis	621	0.25%
End-stage renal disease	138	0.06%

Note. For brevity, we only summarize those hierarchical condition categories that rated as top 10 variables in our analysis.

all five inpatient diagnoses categories, and four HCC codes: metastatic cancer, multiple sclerosis, end-stage renal disease, and stem cell transplant status/complications. The super learner performed better than all the single algorithms included in the analysis of the Truven MarketScan data. Efficiency losses for the single algorithms compared to super learner, with respect to cross-validated R^2, ranged from 4 to 92 percent. Neural net using the top 10 variables from the random forests screening step was the worst performing algorithm, with a relative efficiency of 8 percent. The neural net with all 86 variables also performed poorly, with 15 percent relative efficiency compared to super learner. The parametric regression, lasso, elastic net, ridge, and random forests with all variables performed equivalently, capturing 96 percent of the efficiency of the super learner with cross-validated R^2 values of 0.25. Any of these five algorithms could be chosen as the discrete super learner in practice given the minor absolute differences in performance, although the ridge regression had the best performance. The top 10 versions of these algorithms had a drop in relative efficiency compared to their respective full versions, with 88 percent

Table 3: Results Summary for MarketScan Risk Adjustment Algorithms

Algorithm	$CV\,R^2$	*RE*
Super learner	0.26	–
Parametric regression	0.25	0.96
Top 10	0.23	0.88
Lasso	0.25	0.96
Top 10	0.23	0.88
Elastic net	0.25	0.96
Top 10	0.23	0.88
Ridge	0.25	0.96
Top 10	0.23	0.88
Single tree	0.19	0.73
Top 10	0.19	0.73
Random forests	0.25	0.96
Top 10	0.23	0.88
Neural net	0.04	0.15
Top 10	0.02	0.08

$RE = CV\,R^2(\text{algorithm})/CV\,R^2(\text{super learner})$.

relative efficiency compared to the super learner and cross-validated R^2 values of 0.23. These five top 10 versions all had 92 percent relative efficiency compared their full versions. Both versions of the single tree algorithm had relative efficiencies of 73 percent compared to the super learner. See Table 3 for cross-validated R^2 and relative efficiency values. We present the top 10 parametric linear regression formula coefficients and standard errors in Table 4, as it may be of greatest interest to readers familiar with standard risk adjustment.

Discussion

We introduced cross-validated machine learning methods for prediction in risk adjustment in the Truven MarketScan database, generating new prediction functions, including parsimonious versions of each method. Applying a machine learning framework can be a useful tool for risk adjustment, and it provides researchers with alternatives to large parametric regressions with ever increasing numbers of covariates, which may not provide the flexibility necessary in the age of "big data." When additional novel estimators for prediction are developed, they can easily be added to the ensembling framework described here, as potential candidate learners. Ensembling can augment our learning from data and provide statistical guarantees that we are leveraging the information collected in the strongest possible way. Researchers need not

Table 4: Coefficients and Standard Errors from Top Ten Linear Regression

Variable	*Coefficient (SE)*
Intercept	3,982* (37)
Age, years	
$20 < x \leq 34$	−2,104* (73)
Inpatient diagnoses	
Heart disease	16,453* (246)
Cancer	30,812* (333)
Diabetes	9,000* (320)
Mental health	7,044* (243)
Other	9,382* (138)
Hierarchical condition categories	
Metastatic cancer	44,199* (806)
Stem cell transplant/complication	98,563* (2,801)
Multiple sclerosis	30,058* (623)
End-stage renal disease	129,822* (1,325)

*Significant p-values <.001.

spend energy guessing which algorithm might perform the best or which variables should be included; they can now use super learning to run many at once. The super learner here had the best overall performance. One should note that algorithms that performed well here will not necessarily perform well in other settings, as has been seen in other literature (van der Laan, Polley, and Hubbard 2007; van der Laan and Rose 2011).

Our results also provide preliminary evidence that the use of a lesser number of variables in risk adjustment could actually lead to *better* plan payment risk adjustment. Even examining only the two parametric linear regressions considered within the super learner, the regression with 10 variables had a cross-validated R^2 of 0.23 versus 0.25 when compared to the regression with all 86 covariates. While there is an efficiency loss with respect to cross-validated R^2, it is relatively minor. It is possible that potential cost savings due to the inability to game the risk adjustment system as aggressively as is possible now with the current large number of diagnostic condition codes included in risk adjustment formulas could leave this difference negligible. Thus, even if a full super learner is not performed, a discrete super learner selecting among a number of parametric linear regressions can lead to nontrivial improvements. A discrete super learner could also be designed such that use of the full set of risk adjustment variables would only be warranted if, say, there was a 20 percent loss of efficiency when using the reduced set.

More broadly, deciding if a risk adjustment formula is better requires more extensive empirical evaluation. As noted earlier, a statistical fit measure is the common starting point for evaluation, with a natural next step being simulation-based measures also applied in the literature. It is common, for example, to construct predictive ratios, which compares the predicted payments and costs for subgroups of the population who are regarded as being vulnerable to underservice by plans (Kautter et al. 2014). Parsimony may come at a cost in these terms—by employing a stripped-down empirical model, some disease groups that merit higher payment in conventional risk adjustment models may be underpaid with a simpler model. Any tradeoff here requires empirical work in the context of a particular policy application. Simulation is also needed to fully capture other plan payment features, such as consumer premiums or reinsurance that also affect plan payments and plan incentives (McGuire et al. 2014). Given the impressive fit of the parsimonious model estimated here, consideration of its properties on additional policy-related criteria is clearly merited.

One may notice that our top 10 parametric linear regression does not contain sex. If there are specific variables that *must* be included for important policy reasons, the super learner framework also allows the user to prespecify these variables such that they are included in all algorithms regardless of their results in any screening step. For that matter, different subsets of covariates need not be selected in an automated fashion via a screening step. Policy makers, clinicians, and actuaries can work collaboratively to define subsets based on various considerations (e.g., regulations, sensitivity to upcoding, clinical pathways) and then compare the cross-validated results across these different subsets. One may also be interested in performing cross-validation among different plans to better understand the generalizability of the risk-adjustment formula, or considering plan type as a covariate in the prediction function given the role contracts may play in utilization.

It is important to note that there is increased computing time and memory required in implementing ensemble super learning compared to standard regression techniques. In our paper we also used a number of algorithms that possibly could not be implemented in some settings given both time and computing constraints. However, it may be feasible in those settings to compare a minimal set of regressors, such as a parametric regression with the full set of risk adjustment variables and one with 10 variables. The cross-validation of the two algorithms would be performed, selecting the optimal regression based on cross-validated *MSE* or R^2 or other predefined rule balancing formula complexity and performance, and then returning a final fixed regression formula fit on the full data for use in new data. The key point being that our study provides supporting evidence that using an exhaustive set of variables in a parametric linear regression for risk adjustment, as is standard practice, may not be necessary compared to a simpler formula: a regression with just 10 variables.

While we focused here on the estimators and machine learning techniques that can be employed in large claims databases and surveys, it is also important to look toward the future. We are quickly heading toward a health care structure where the amount of data outgrows the current capabilities of our research systems, and as noted above, in some settings this is already the case. We must consider how these new prediction methods can be integrated practically into risk adjustment systems. Over time, this will include the development of native big data systems that combine rigorous statistical methods, the software to analyze the data, and databases that allow for rapid discovery. While plan payment risk adjustment was introduced to combat the issue of consumer selection into insurance plans, the implementation of cross-validated machine learning and more parsimonious formulas has yet to be incorporated, and it may provide improvements over current procedures.

ACKNOWLEDGMENTS

Joint Acknowledgment/Disclosure Statement: This work was supported by the John and Laura Arnold Foundation, NIMH R01-MH094290, and an NIH-NIA pilot grant from the Program on the Global Demography of Aging at the Harvard T.H. Chan School of Public Health. The author thanks Thomas McGuire, Timothy Layton, Randall Ellis, and the anonymous reviewers for helpful comments on an earlier version of this manuscript.

Disclosures: None.

Disclaimers: None.

REFERENCES

Adamson, D. M., S. Chang, and L. G. Hansen. 2008. *Health Research Data for the Real World: The MarketScan Databases.* New York: Thompson Healthcare.

Breiman, L., J. Friedman, C. J. Stone, and R. A. Olshen. 1984. *Classification and Regression Trees.* New York: CRC Press.

Breyer, F., K. Bundorf, and M. V. Pauly. 2012. "Health Care Spending Risk, Health Insurance, and Payment to Health Plans." In *The Handbook of Health Economics*, Vol. 2, edited by M. Pauly, T. McGuire, and P. Barros, pp. 691–792. Amsterdam: Elsevier.

Brown, J., M. Duggan, I. Kuziemko, and W. Woolston. 2014. "How Does Risk Selection Respond to Risk Adjustment? Evidence from the Medicare Advantage Program." *American Economic Review* 104 (10): 3335–64.

Dudoit, S., and M. J. van der Laan. 2005. "Asymptotics of Cross-Validated Risk Estimation in Estimator Selection and Performance Assessment." *Statistical Methodology* 2 (2): 131–54.

Einav, L., and A. Finkelstein. 2011. "Selection in Insurance Markets: Theory and Empirics in Pictures." *Journal of Economic Perspectives* 25 (1): 115–38.

Einav, L., A. Finkelstein, and M. R. Cullen. 2010. "Estimating Welfare in Insurance Markets Using Variation in Prices." *Quarterly Journal of Economics* 125 (3): 877–921.

Friedman, J., T. Hastie, and R. Tibshirani. 2001. *The Elements of Statistical Learning.* New York: Springer.

———. 2010. "Regularization Paths for Generalized Linear Models via Coordinate Descent." *Journal of Statistical Software* 33 (1): 1–22.

Geruso, M., and T. Layton. 2015. "Upcoding: Evidence from Medicare on Squishy Risk Adjustment." NBER Working Paper 21222 [accessed on June 1, 2015]. Available at http://www.nber.org/papers/w21222

Glazer, J., and T. G. McGuire. 2000. "Optimal Risk Adjustment of Health Insurance Premiums: An Application to Managed Care." *American Economic Review* 90 (4): 1055–71.

Iezzoni, L. 2012. *Risk Adjustment for Measuring Healthcare Outcomes*, 4th Edition. Chicago, IL: Health Administration Press.

James, G., D. Witten, T. Hastie, and R. Tibshirani. 2013. *An Introduction to Statistical Learning.* New York: Springer.

Kautter, J., G. C. Pope, M. Ingber, S. Freeman, L. Patterson, M. Cohen, and P. Keenan. 2014. "The HHS-HCC Risk Adjustment Model for Individual and Small Group Markets under the Affordable Care Act." *Medicare & Medicaid Research Review* 4 (3).

Kronick, R., and W. P. Welch. 2014. "Measuring Coding Intensity in the Medicare Advantage Program." *Medicare & Medicaid Research Review* 4 (2).

van der Laan, M. J., E. C. Polley, and A. E. Hubbard. 2007. "Super Learner." *Statistical Applications in Genetics and Molecular Biology* 6: 25.

van der Laan, M. J., and S. Rose. 2011. *Targeted Learning: Causal Inference for Observational and Experimental Data.* New York: Springer.

Layton, T., R. Ellis, and T. McGuire. 2015. "Assessing Incentives for Adverse Selection in Health Plan Payment Systems." NBER Working Paper 21531 [accessed on October 1, 2015]. Available at http://www.nber.org/papers/w21531

Lee, B., J. Lessler, and E. A. Stuart. 2009. "Improving Propensity Score Weighting Using Machine Learning." *Statistics in Medicine* 29: 337–46.

Liaw, A., and M. Wiener. 2002. "Classification and Regression by RandomForest." *R News* 2 (3): 18–22.

McGuire, T. G., J. P. Newhouse, S.-L. Normand, J. Shi, and S. Zuvekas. 2014. "Assessing Incentives for Service-Level Selection in Private Health Insurance Exchanges." *Journal of Health Economics* 35: 47–63.

Polley, E., and M. J. van der Laan. 2013. *SuperLearner: Super Learner Prediction. R Package Version 2.0-10.*

R Core Team. 2013. *R: A Language and Environment for Statistical Computing.* Vienna, Austria: R Foundation for Statistical Computing.

Rose, S. 2013. "Mortality Risk Score Prediction in an Elderly Population Using Machine Learning." *American Journal of Epidemiology* 177 (5): 443–52.

Rose, S., J. Shi, T. McGuire, and S. L. Normand. 2015. "Matching and Imputation Methods for Risk Adjustment in the Health Insurance Marketplaces." *Statistics in Biosciences.* doi:10.1007/s12561-015-9135-7.

Sudat, S. E., E. J. Carlton, E. Y. Seto, R. C. Spear, and A. E. Hubbard. 2010. "Using Variable Importance Measures from Causal Inference to Rank Risk Factors of Schistosomiasis Infection in a Rural Setting in China." *Epidemiologic Perspectives & Innovations* 7: 3.

Therneau, T., B. Atkinson, and B. Ripley. 2013. *rpart: Recursive Partitioning. R package version 4.1-3.*

van Veen, S. H. C. M., R. C. van Kleef, W. P. M. M. van de Ven, and R. C. J. A. van Vliet. 2015. "Is There One Measure of Fit That Fits All? A Taxonomy and Review of Measures of Fit for Risk Equalization Models." *Medical Care Research and Review* 72 (2): 220–43.

Venables, W. N., and B. D. Ripley. 2002. *Modern Applied Statistics with S*, 4th Edition. New York: Springer.